

*Exploiting Percolation and Randomness in the Concurrent Logical  
and Physical Design of Digital Nanoscale Circuits*

FENA Phase III – White Paper

*Principal Investigator:*

**Marc Riedel**, Ph.D.

Assistant Professor,

Electrical & Computer Engineering;

Graduate Faculty,

Biomedical Informatics & Computational Biology;

University of Minnesota.

address: 200 Union St. S.E., Minneapolis, MN 55455

email: [mriedel@umn.edu](mailto:mriedel@umn.edu)

tel: 612-625-6086

cell: 612-275-9878

fax: 612-625-4583

web: <http://www.cctbio.ece.umn.edu>

*Contracts and Grants Officer:*

**Kevin McKoskey**, CRA

Senior Grants Manager

Sponsored Projects Administration

University of Minnesota

address: 200 Oak Street S.E., Minneapolis, MN 55455

email: [kevin@umn.edu](mailto:kevin@umn.edu)

tel: 612-624-5066

fax: 612-624-4843

## *Exploiting Percolation and Randomness in the Concurrent Logical and Physical Design of Digital Nanoscale Circuits*

### 1 State of the Art

The successful design formula for digital circuits has been rigidly hierarchical, with sharp boundaries in the abstractions at different levels. At the logic level, there are standardized gates and components with prescribed functionality. At the system level, there are microprocessors with common instruction sets. At the algorithmic level, software executes without explicit reference to the underlying hardware. At all levels, there exists a substantial investment in the CAD tools – indeed, also in the expertise of the end-users – tailored for specific design and optimization tasks.

It is between the logical level and the physical level that the classic analog/digital boundary occurs: beneath it, circuits are physical devices operating on real-valued voltage signals; above it, they process zeros and ones. Implicitly, there is another boundary that occurs here, that between physical modeling and functionality: above it there is certainty and determinism; beneath it, there is uncertainty and randomness.

Indeed, while physical circuits are modeled with variances and designed with error margins, logical circuits are not. From the logic level up, the precise Boolean functionality of a circuit is fixed; it is up to the physical layer to produce voltage values that can be interpreted as the exact logical values that are called for. This abstraction is firmly entrenched yet costly: variability, uncertainty, noise – all must be compensated for through ever more complex design techniques at the physical level. As the scale of technology pushes into ever deeper regimes, the cost and complexity are threatening to stall progress. We argue that the deterministic paradigm for computation is untenable.

In our work in phase II of FENA, we have been developing a general methodology for **synthesizing stochastic logic**, that is to say, digital circuitry that processes signals probabilistically, and so can cope with errors and uncertainty. The method synthesizes both combinational and sequential circuitry. It implements polynomial functions, a category that is important for data-intensive applications such as digital signal processing; it synthesizes non-polynomial functions through polynomial approximations (MacLaurin series expansions). The resulting logic processes serial or parallel streams that are random at the bit level. However, in the aggregate, the computation becomes *accurate*, since the results depend only on the statistics, not on specific bit values.

The approach provides fault tolerance, both for the underlying logic components as well as for the signal values. Experiments show that our method produces circuits that are **highly tolerant of errors such as noise-induced bit flips** (see Figure 1). Unlike all previous methods, this tolerance of transient faults scales gracefully to very large numbers of errors.

### 2 Summary of Objectives

Our broad objective is to develop computational techniques for logic synthesis for emerging nanoscale materials and processes. We have identified **probabilistic logic** and **stochastic information processing** as key concepts. In FENA phase III, we will *focus* the task of synthesizing probabilistic logic more narrowly and we will also *generalize* it:

- We will apply a novel synthesis technique for implementing robust digital computation in nanofabrics with random interconnects: **percolation theory on random graphs**. We will exploit the non-linearity that occurs through percolation to produce Boolean outputs from input probabilities. We will achieve this computation with no switches or logic elements: the functionality is produced by the percolation activity that occurs due to the random connectivity of the fabric. Also, we will apply so-called *Bernstein multiplexing* for the synthesis of logic with **probabilistic wire bundles**.

- We will develop a vertically-integrated design flow for probabilistic computation, from the level of logic through to the system architecture. As a proof of concept, we will synthesize a complete open-RISC processor in the stochastic paradigm. We will evaluate its performance and reliability through extensive simulation and benchmarking.

### 3 Technical Approach

Consistent with its forward-looking and long-term mandate, FENA is developing a wide range of materials, devices, and physical computing structures as possible replacements for CMOS. Most FENA technologies are in the exploratory phases, still years or decades from the point when they will be actualized. Accordingly, the development of tools and techniques for logic synthesis remains speculative. And yet, it has become clear that the path to scaling beyond CMOS will not end with new *physical structures*, such as switches and logic gates. FENA technologies will pose significant challenges for the *logical design* of circuits as well.

We can identify broad traits that will likely impinge upon logic synthesis. On the one hand, FENA technologies will provide unprecedented densities of bits, switches, and logic gates. On the other hand, the topology of circuits that are produced at the nanoscale, for instance, through self-assembly, is typically constrained and regular. The circuits are often characterized by high defect rates and spurious connections; in some cases there is inherent randomness in the interconnects. These key features motivate the two architectures that we consider for synthesis: **nanofabrics with random interconnects** and **probabilistic wire bundles**.

#### 3.1 Logic Synthesis Through Percolation

The project will develop a synthesis strategy for nanofabrics based upon the phenomenon of percolation in random graphs. Percolation theory is a rich mathematical topic that forms the basis of explanations of physical phenomena such as diffusion and phase changes in materials. It tells us that in media with random local connectivity, there is a critical threshold for global connectivity: below the threshold, the probability of global connectivity quickly drops to zero; above it, the probability quickly rises to one. This is illustrated in Figure 2. We will exploit the theory in a novel way: we will use the *non-linearity* produced by percolation to implement digital computation in regular nanofabrics.

The approach is illustrated on a nanofabric consisting of a large crossbar array in Figure 3. The input values are applied to distinct regions in the array. In each region, the probability of local connectivity is a linear function of the corresponding input value.

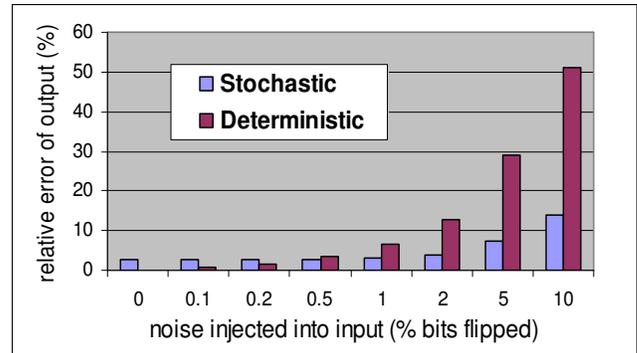


Figure 1: **The error tolerance of deterministic vs. stochastic computation on evaluations of polynomials.** Sixth-order Maclaurin polynomial approximations of trigonometric functions ( $\sin(x)$ ,  $\tan(x)$ ,  $\arcsin(x)$ ,  $\arctan(x)$ , etc.) with 10 bit resolution were synthesized. The error ratio is the fraction of random bit flips that are injected into the input data. The figure shows the relative number of bit errors in the output.

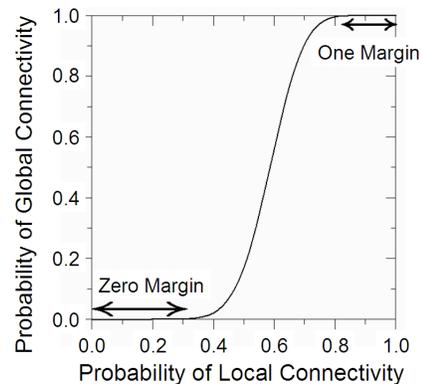


Figure 2: **Non-linearity Through Percolation in Random Media.** Percolation produces a non-linear relationship between the probability of local connectivity and the probability of global connectivity. The sharper the non-linearity, the higher the quality of the computation. The relationship between input values and the probability of local connectivity can be any function, linear or non-linear.

Due to percolation, the probability of connectivity across each region is essentially zero or one, depending on whether the corresponding input value is above or below the critical threshold. We use black and white squares to illustrate this (Figure 4). Connectivity through the entire array, from the plate at the top to the plate at the bottom, is now a Boolean function of the input values via the linear relationship between the input values and the probability of local connectivity. The Boolean function evaluates to 1 if there exists a path from the top plate to the bottom plate, and evaluates to zero otherwise. With this scheme, robust digital computation is achieved without switches or logic elements.

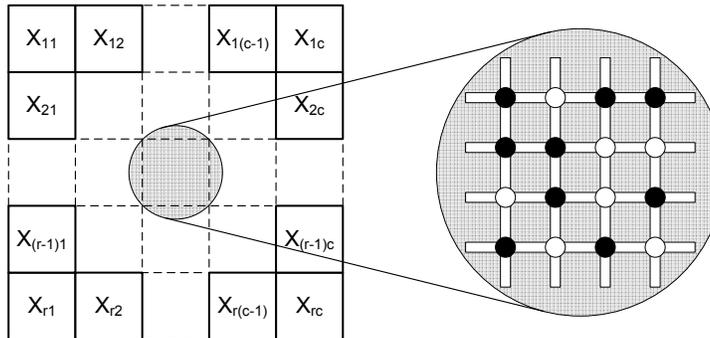


Figure 3: **Using a Large Crossbar Array as the Nanofabric.** There are  $r \times c$  input values,  $X_{11}, \dots, X_{rc}$ , each applied to regions of the nanofabric. Each input value controls the probability that connections form between horizontal and vertical nanowires in the corresponding region.

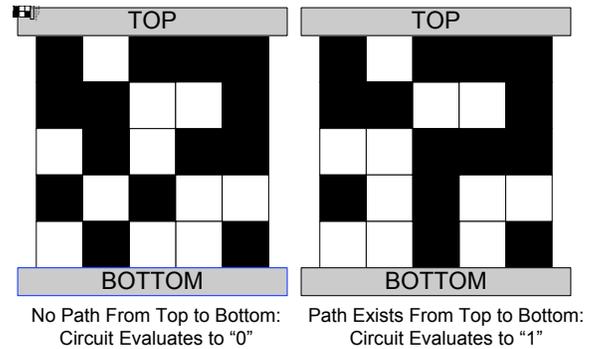


Figure 4: **Boolean Computation with Percolation.** Due to percolation, the probability of connectivity across each region is essentially zero (a white square) or one (a black square). The nanofabric implements a Boolean function that evaluates to one if there is global path through the entire array (from top to bottom) and zero if there is no such path.

An important consideration in the synthesis methodology is the quality of the **noise margins**. The zero and one margins are the ranges of input values where logic 0 and logic 1 are achieved unambiguously. Different assignments of input variables to regions of the nanofabric affect these margins. In particular, assignments that evaluate to 0 but have diagonally adjacent assignments of blocks of 1's result in poor zero margins. This is illustrated in Figure 7. Note that each column corresponds to the AND of the inputs; the columns taken together correspond to the OR of these AND terms (so implement a “sum-of-products” expression). This interpretation gives rise to an interesting problem in logic optimization: given a Boolean sum-of-products expression as the target function, how should one assign the variables to the regions of the fabric such that there are no diagonally adjacent variables that are 1 in any assignment that evaluates to 0.

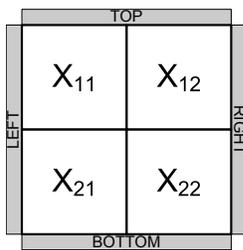


Figure 5: A nanofabric with four input values,  $X_{12}, X_{21}, X_{12}$ , and  $X_{22}$ .

$X_{11}$	$X_{21}$	$X_{12}$	$X_{22}$	Boolean Function	Noise Margins
0	0	0	0	0	40%
0	0	0	1	0	25%
0	0	1	1	1	14%
0	1	0	1	0	23%
0	1	1	0	0	0%
0	1	1	1	1	14%
1	1	1	1	1	25%

Figure 6: Possible 0/1 assignments to the inputs in Figure 5 (up to symmetries) and the corresponding noise margins.

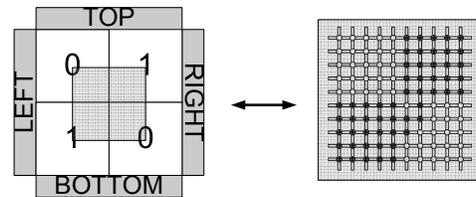


Figure 7: An input assignment with a low noise margin. Here diagonally adjacent blocks of 1's result in a poor zero margin: there is a good chance that a weak a connection from top to bottom will form through stray, random connections across the diagonal.

### 3.2 Logic Synthesis for Probabilistic “Bundles”

This project advocates a novel view for computation: instead of synthesizing circuits that transform definite inputs into definite outputs – say, Boolean, integer, or real values into the same – we synthesize circuits that transform probability values into probability values; so, conceptually, real-valued probabilities are both the inputs and the outputs. Instead of computing with deterministic signals, operations at the logic level are performed on randomized values in serial streams or in parallel “bundles” of wires. When serially streaming, the signals are probabilistic in *time*; in parallel, they are probabilistic in *space*. Figure 8 illustrates the latter. The wires in the bundles are digital, carrying 0’s and 1’s; they are processed by ordinary logic gates, such as AND and OR. However, the signal is conveyed through the *statistical distribution* of the logical values. Real values in the interval  $[0, 1]$  correspond to the fraction of wires with value 1. For example, if two-fifths are 1, then the signal is  $p = 0.4$ .

With physical uncertainty, the fractional numbers correspond to *probability of occurrence* of logical 1 versus logical 0 in an observation interval. In this way, computations in the **deterministic Boolean domain** are transformed into **probabilistic computations in the real domain**. Our synthesis strategy is to cast logic computations as arithmetic operations in the probabilistic domain and implement these directly as stochastic operations on data-paths.

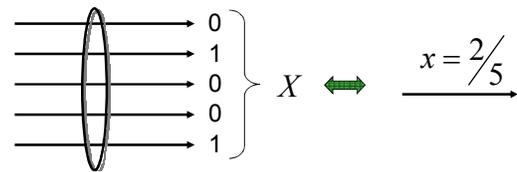


Figure 8: **A Stochastic Bundle**: a real value  $x$  in  $[0, 1]$  is represented as a bundle of bits  $X$ . For each bit, the probability that it is one is:  $P(X = 1) = x$ .

## 4 Statement of Work

We will create a comprehensive framework for probabilistic logic and stochastic information processing with emerging FENA technologies. Two Ph.D. students, Weikang Qian and Mustafa Altun, and an undergraduate, John Backes, in ECE at the University of Minnesota will form the research team. All research will be validated through benchmarking on industrial designs and detailed performance measurements. Close collaboration with the SRC member companies is expected, particularly with respect to defining and refining the goals as well as in guiding the validation efforts.

1. **FENA Phase III, Year 1:** A full synthesis methodology for stochastic logic will be developed for both architectures:
  - (a) Percolation in nanofabrics with random interconnects. (*6 months*)
  - (b) Stochastic multiplexing in bundled nanowire arrays. (*6 months*)
2. **FENA Phase III, Year 2:**
  - (a) Synthesis will be integrated in a common software platform. Extensive benchmarking and simulations will be performed on the IWLS benchmark set. (*6 months*)
  - (b) The resilience of designs to **transient** and **permanent faults** will be quantified; also the **area**, **delay**, and **power dissipation** will be measured. (*6 months*)
3. **FENA Phase III, Year 3:**
  - (a) A complete open-RISC processor with stochastic datapaths and memory subsystems will be synthesized using each of our methodologies. Its performance, accuracy, and reliability will be evaluated. (*6 months*)
  - (b) Architectural optimizations that are tailored to specific application domains – in particular applications that are data-intensive yet statistical in nature, such as scientific computing – will be explored. (*6 months*)

## **Biography of the Principal Investigator:**

Please see <http://www.cctbio.ece.umn.edu>